# Interrupts of 8085

## Interrupt

An external I/O signal or an instruction can suspend normal flow of execution and go to ISR. Once serviced, program resumes.

## ISR

A small program or a routine that when executed services the corresponding interrupting source is called as an ISR.

## What happens when interrupt comes?

- INTA (active low) signal to the peripheral.

- Vector address of particular interrupt is stored in program counter.

- The processor executes an ISR.

# Interrupts in 8085 microprocessor

When microprocessor receives any interrupt signal from peripheral(s) which are requesting its services, it stops its current execution and program control is transferred to a sub-routine by generating **CALL** signal and after executing sub-routine by generating **RET** signal again program control is transferred to main program from where it had stopped.

When microprocessor receives interrupt signals, it sends an acknowledgement (INTA) to the peripheral which is requesting for its service.

Interrupts can be classified into various categories based on different parameters:

## Hardware and Software Interrupts

When microprocessors receive interrupt signals through pins (hardware) of microprocessor, they are known as *Hardware Interrupts*. There are 5 Hardware Interrupts in 8085 microprocessor. They are – *INTR, RST 7.5, RST 6.5, RST 5.5, TRAP. Software Interrupts* are those which are inserted in between the program which means these are mnemonics of microprocessor. There are 8 software interrupts in 8085 microprocessor. They are – *RST 0, RST 1, RST 2, RST 3, RST 4, RST 5, RST 6, RST 7.*

# Vectored and Non-Vectored Interrupts

*Vectored Interrupts* are those which have fixed vector address (starting address of sub-routine) and after executing these, program control is transferred to that address.

Vector Addresses are calculated by the formula 8 * TYPE

| Interrupt | Vector Address |
|---|---|
| TRAP (RST 4.5) | 24 H |
| RST 5.5 | 2C H |
| RST 6.5 | 34 H |
| RST 7.5 | 3C H |

**For Software interrupts vector addresses are given by:**

| Interrupt | Vector Address |
|-----------|----------------|
| RST 0 | 00 H |
| RST 1 | 08 H |
| RST 2 | 10 H |
| RST 3 | 18 H |
| RST 4 | 20 H |
| RST 5 | 28 H |
| RST 6 | 30 H |
| RST 7 | 38 H |

*Non-Vectored Interrupts* are those in which vector address is not predefined. The interrupting device gives the address of sub-routine for these interrupts. *INTR* is the only non-vectored interrupt in 8085 microprocessor.

## Maskable and Non-Maskable Interrupts

- *Maskable Interrupts* are those which can be disabled or ignored by the microprocessor. These interrupts are either edge-triggered or level-triggered, so they can be disabled. *INTR, RST 7.5, RST 6.5, RST 5.5* are maskable interrupts in 8085 microprocessor.

- Non-Maskable Interrupts are those which cannot be disabled or ignored by microprocessor. *TRAP* is a non-maskable interrupt. It consists of both level as well as edge triggering and is used in critical power failure conditions.

# Priority of Interrupts

When microprocessor receives multiple interrupt requests simultaneously, it will execute the interrupt service request (ISR) according to the priority of the interrupts.
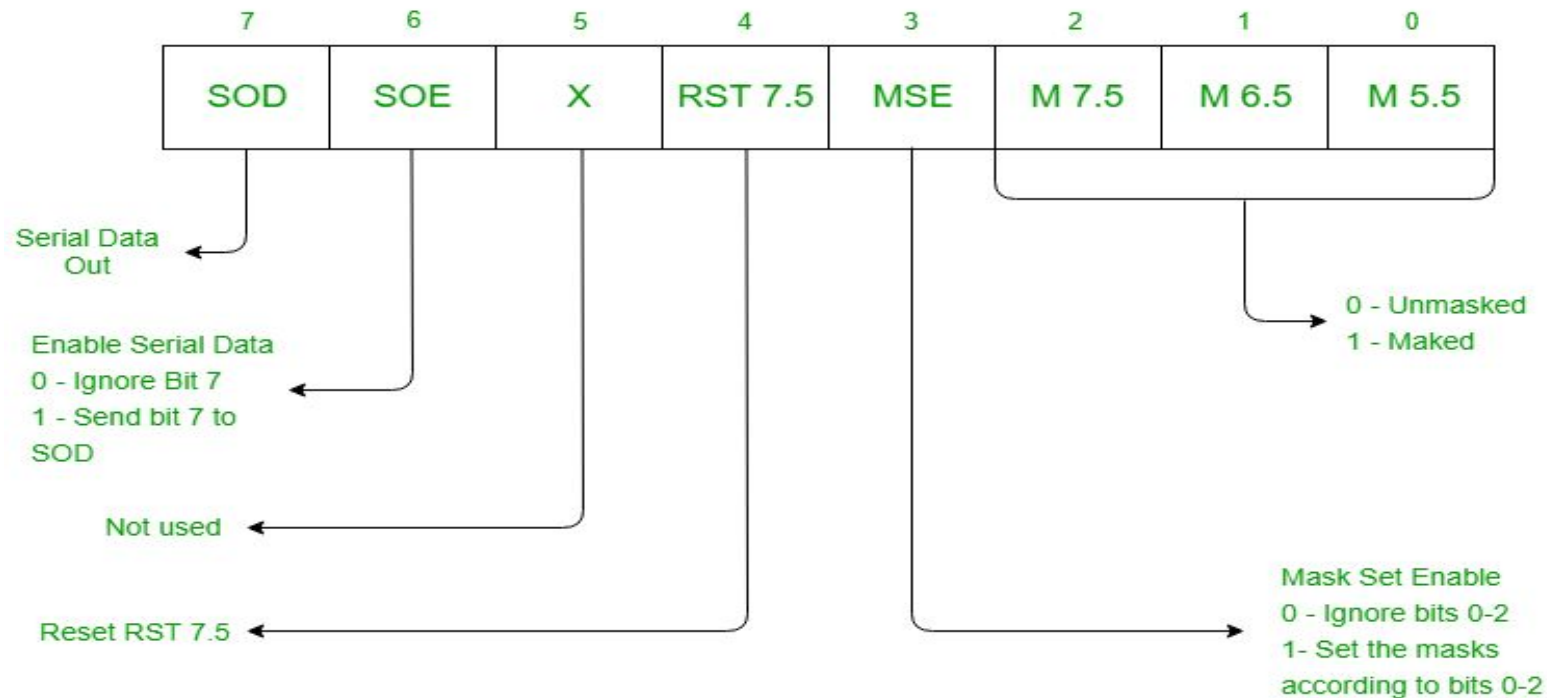
TRAP

RST 7.5

RST 6.5

RST 5.5

INTR

Highest

Lowest

# Instruction for Interrupts

**Enable Interrupt (EI) –** The interrupt enable flip-flop is set and all interrupts are enabled following the execution of next instruction followed by EI. No flags are affected. After a system reset, the interrupt enable flip-flop is reset, thus disabling the interrupts. This instruction is necessary to enable the interrupts again (except TRAP).

**Disable Interrupt (DI) –** This instruction is used to reset the value of enable flip-flop hence disabling all the interrupts. No flags are affected by this instruction.

**Set Interrupt Mask (SIM) –** It is used to implement the hardware interrupts (RST 7.5, RST 6.5, RST 5.5) by setting various bits to form masks or generate output data via the Serial Output Data (SOD) line. First the required value is loaded in accumulator then SIM will take the bit pattern from it.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SOD | SOE | X | RST 7.5 | MSE | M 7.5 | M 6.5 | M 5.5 |

Serial Data
Out

Enable Serial Data
0 - Ignore Bit 7
1 - Send bit 7 to
SOD

Not used

Reset RST 7.5

0 - Unmasked
1 - Maked

Mask Set Enable
0 - Ignore bits 0-2
1- Set the masks
according to bits 0-2

**Read Interrupt Mask (RIM) –** This instruction is used to read the status of the hardware interrupts (RST 7.5, RST 6.5, RST 5.5) by loading into the A register a byte which defines the condition of the mask bits for the interrupts. It also reads the condition of SID (Serial Input Data) bit on the microprocessor.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SID | P 7.5 | P 6.5 | P 5.5 | IE | M 7.5 | M 6.5 | M 5.5 |

Serial Data In

0 - No request pending
1 - Interrupt request pending

0 - Unmasked
1 - Maked

Value of Interrupt Enable flip-flop